Report Transfer Learning of Deep Convolutional Network on Twitter

Hoa T Le, Christophe Cerisara, Alexandre Denis

Report Transfer Learning of Deep Convolutional Network on Twitter

Hoa T. LE [1], Christophe Cerisara [1], Alexandre Denis [2]
[1] LORIA, UMR 7503, Nancy, France; [2] SESAMm, France

July 13, 2017

Abstract

This report aims at showing the capacity of transfering a *deep neural network* on *char-level* on massive dataset Twitter, using *distant supervision*. We showed that more data could help for the Stanford140 dataset. The best overal result observed is 84% of transfer learning for two sentiment polarity classes (positive-negative) from 16M emoticons subjective SESAMm dataset to small SemEval 2013 dataset. Other learning of three classes (with neutral) or nine classes based on emoticons (happy, laughing, kisswink, playful, sad, horror, shock, annoyed, hesitated) didn't show any advantages yet in the study.

1 Motivation and Previous Work

The recent succes of deep learning methods comes with a heavy dependence on the massive label data set [8], [6]. However, there are a lot of domains where the need of human supervised data could become exponentially expensive. As for the case of Twitter, the number of human-labeled data could never catch up the speed of new creating tweets every day. *Distant supervision* thus gradually become a prominent method for many twitter application tasks. To profit the scale of big data set, this method collect *automatically* a large number of tweets via some pre-defined rules (hashtag, lexicons, emoticons,...) and transfer to another smaller specific tasks. For sentiment analysis, Go et al., (2009) [4] show that emoticons is the most reliable and effective choice. Indeed, many people have followed this stategy and end up winning Semeval competition for many years ([11], [2]). In these works, they transfer only a shallow convolutional networks [7] on word-level but achieve considerable result.

Recently, there is a line of work [10] showing that learning directly *on char-level* on twitter could be a better choice because twitter normally contain a lot of slang, elongated words, contiguous sequences of exclamation marks, abbreviations, hashtags,... For this end, inspiring from computer vision ([8], [5]) Zhang et al., (2015) [13], Conneau et al., (2016) [1] developped a deep network (6 and 29 convolutional layers respectively) to learn directly on atom-level. As realising a deep network could leverage better with a corresponding massive dataset, we want to explore if it could beat the traditional transfer learning of a shallow network on word-level [4], [3].

2 Experimental results

2.1 Sentiment Analysis Data

Stanford140: this consists 1.6 millions tweets, automatically collected in 2009 on a wide range of topics by Twitter Search API. The dataset is balanced with 800k tweets with positive emoticons :) and 800k tweets with

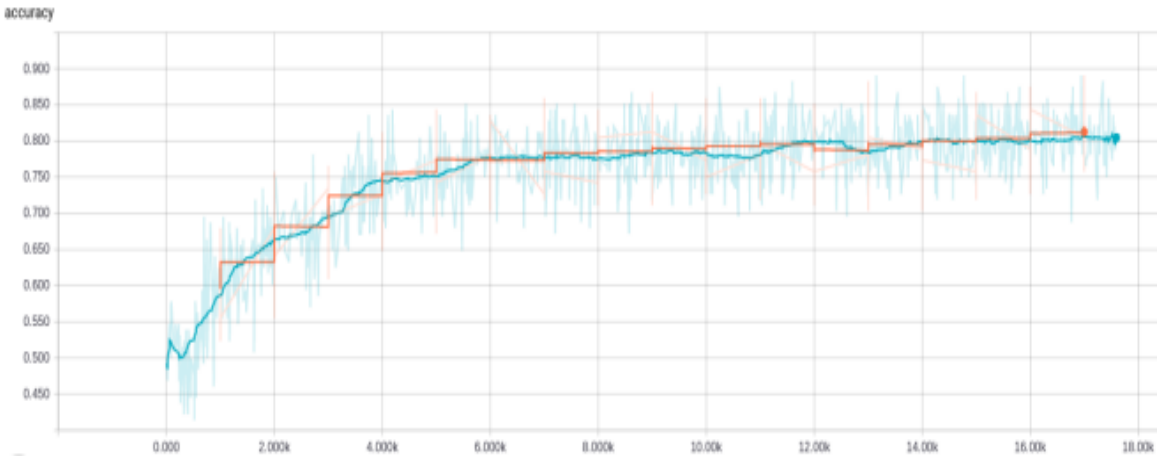


Figure 1: Results on 1.6M training and test set

negative emoticons :(. The training dataset is already pre-processesd: emoticons are stripped off; any tweet containing both positive and negative emoticons are removed; retweets, duplicated tweets and tweets with ":P" are also removed to reduce the most possible the bias in the dataset (more further details of preprocessing could be followed in [4]).
SESAMm data [1]: this consists **3 billion subjective tweets**, collected automatically from the period 2014-2016, without any particular filter of preprocessing emoticons. From this pool, we filter out 16M balanced positive/negative emoticons tweets as the rules of [4], for the ease of comparison. The preprocessing procedure also follow the description above.

2.2 Deep Neural Network on Twitter

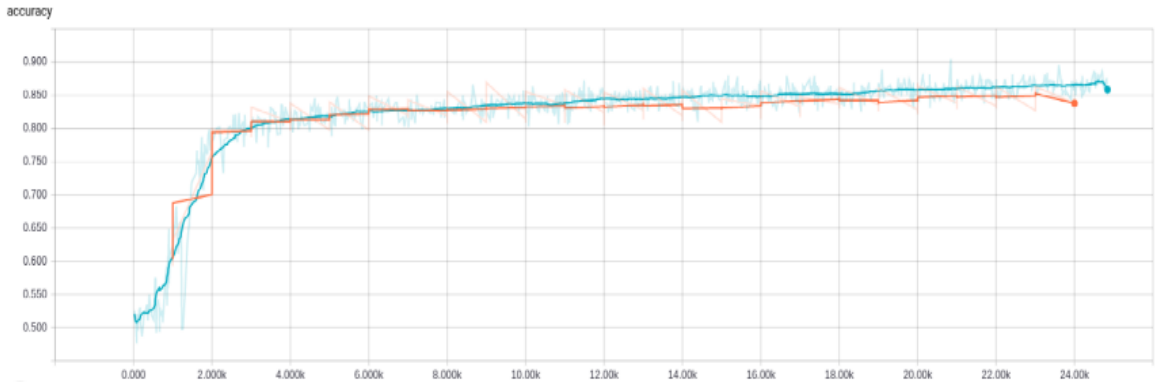
2.2.1 Deep Neural Network

For all the experiements, we explore the structure of deep model on char-level as described in Conneau et al., 2016 [1]. Because the length of the sequence in twitter is shorter than amazon or movie reviews [1], we study only two structures: (1-1-1) and (2-2-1) double-convolutional blocks with (256-256-128) kernel window size correspondingly. Besides, we use five fully connected layer with the configuration (5632-4096-2048-512-128) neurons respectively to observe better the transfer's process. The hyperparameters are followed [1]: Adam Optimizer with an initial learning rate of 0.001, 128 batch size, no L2 regularization and drop-out. Each character is represented as a one-hot encoding vector where the dictionary contains the following 69 tokens: "$abcdefghijklmnopqrstuvwxyz0123456789-, ; .!? : "/|\_\#\%\& \star\prime + = <> ()[]$". The maximum sequence length is 176; smaller texts are padded with 0 while larger texts are truncated. The convolutional layers are initialized following [5]. Training is iterated per steps and we evaluate it at every 1000 steps on test set.

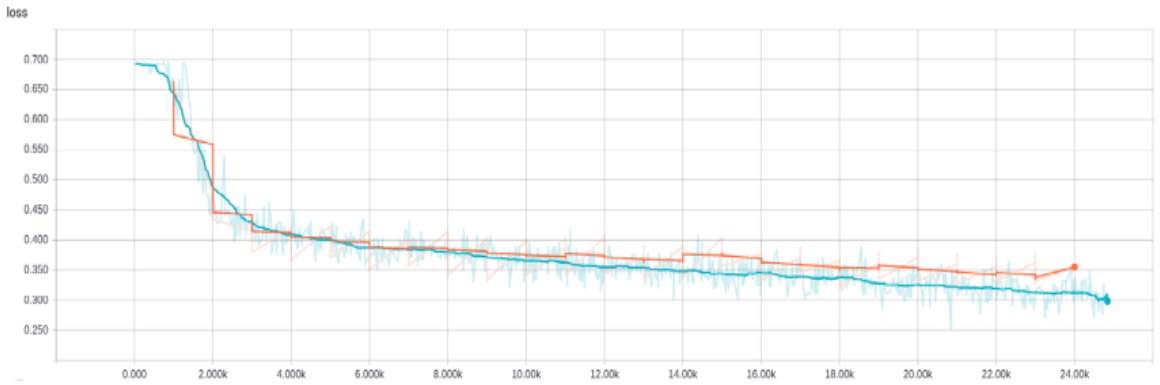
2.2.2 Stanford Emoticon Twitter data

We first apply the model (1-1-1) double convolutional blocks to Stanford140 dataset. To see the benefit of data scale, we evaluate the model on data level: 10k, 60k and 1.6M samples. The results are given in the figure 1 below (and figure 8 in the appendix)

[1] $http://www.sesamm.com/$



(a) Accuracy on 1.943M training and test set



(b) Loss on 1.943M training and test set

Figure 2: Result of augmented Stanford Emoticon Twitter data

As we see in figure 1, it took 14k steps to converge to 80% on both training and test set (note that the shaded line is the real realisation and the bold line is the fitted average, outputted by TensorBoard). This is a good balance between the depth and the quantity of the data. If we use less data (10k and 60k as in figure 8), the model can overfit easily. We have tried dropout on the last fully connected layer but it didn't help. Also, as we observed, taking into account more convolutional blocks (model (2-2-1)) with this size of dataset didn't gain any more performance. We hypothese that a much deeper network will need a massive corresponding dataset to observe the effect, which is our subject of study in section 2.2.3 and 2.2.4.

2.2.3 Augmenting Stanford Emoticon Twitter data

As Stanford Emoticon Twitter data contains very general samples which will help to generalise better, we're interested in augmenting the quantity of the data to see if the model still can get benefit. We've collected by Twitter Streaming 343k positive and negative tweets (by the same emoticons :) :(as Stanford data) from January to March 2017. The result of this augmented data and bigger model (2-2-1) is a gain of 5% more on the test set, which is precisely 85% overall (figure 2). The model still show good compromise between learning and generalization without too much overfitting. The model is then used to compare with model learnt on big (but subjective) SESAMm dataset in section 2.2.4.

2.2.4 SESAMm data

By the constraint of time for collecting new tweets, we can not explore quickly the model on a large scale. However, SESAMm possess already in hands a very large dataset. We're interested if it still can learn some useful patterns differently than the general case and possibly could help our classifier become more discriminative regards each sentiment polarity. The result show interestingly a level 85% accuracy performance on test set, as the augmented Stanford dataset case. We suspect that this is nearly the best performance we could achieve with distant supervision for two classes. It seems that there's some threshold to determine if putting more data in could help. Generally, the first part of data (the first few millions observations) could set up for the whole performance and the later (dozen millions of observations) could barely or couldn't help at all.

2.3 Transfer Learning

In this section, we carry out the transfer learning for two case: augmenting Stanford dataset (2.2.3) and SESAMm data (2.2.4). The transfer was done on the Semeval 2013 training and test set.

The transfer from augmenting Stanford data result in a 75% on Semeval 2013 test set (figure 3) if only fine-tune the last layer. As the training set of Semeval is small, we see that the learning could overfit easily without using dropout. Fine-tune more fully connected layers (2 or 3 last layers) degrade more and more the performance (figure 9). This is coherent with what is observed in the vision domain ([12]), where the very last layer contains very specific feature and the first layers contain very generic feature. Yosinski et al., (2014) [12] suggested that if the target transfer data is small and the domain is close, one should reuse the whole structure and re-train only the last linear classifier on top.

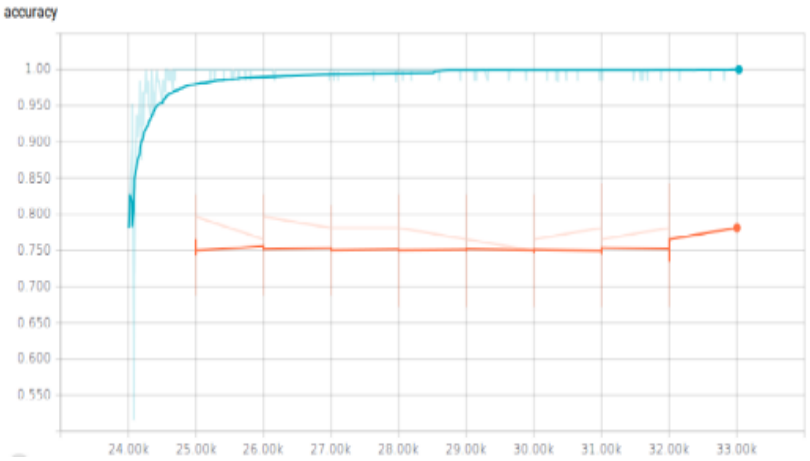


Figure 3: Accuracy of transfer learning from augmenting Stanford Emoticon Twitter data to SemEval 2013 training and test set. Fine tune only last fully connected layer, without dropout

As the nature of SESAMm data is different than those of Stanford data, we're intested in examining the capacity of transfer learning in this case, with dropout on Semeval test set to prevent overfitting. Figure 4 show a net amelioration of 84% on test set. The transfer is showed clearly as the starting point of training and test curve are very high (60% and 75% correspondingly) comparing to without the transfer case. Fine-tune more layer (figure 10) could only lead to overfitting and degrade the performance as observed above.

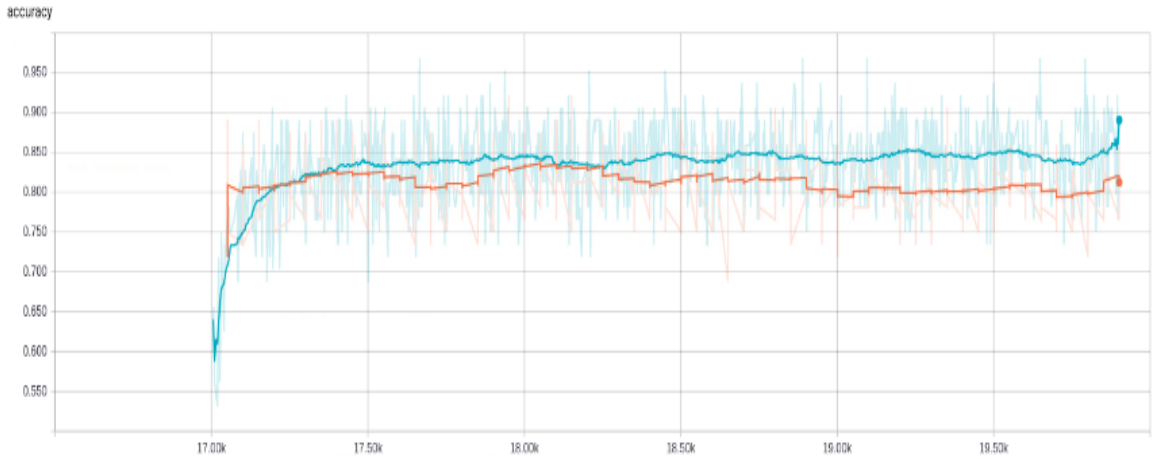
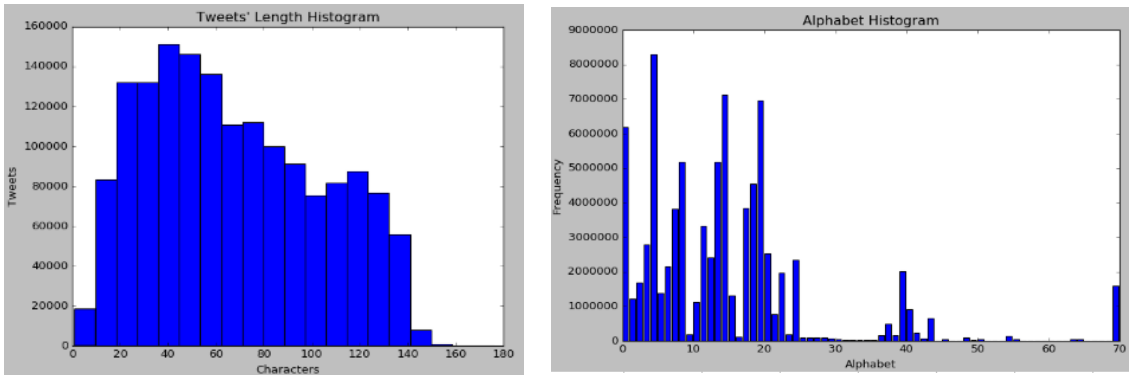


Figure 4: Accuracy of transfer learning from 16M tweets SESAMm data to SemEval 2013 training and test set. Fine tune only last fully connected layer, with dropout



(a) Count of tweet's length

(b) Count of tweet characters

Figure 5: Histogram Distribution of Stanford Emoticon Twitter data

3 Qualitative analysis and other approaches

To verify the effectiveness of chosen hyperparemters (maximum 176 characters and alphabet following [1]), we plot the frequency of tweet's length and each characters of the alphabet as below (figure 5). There are 67 characters on average in each tweets, 239 tweets exceed our limit maximum 176 characters lengths (which will be truncated). Most of these tokens are characters and numbers; there is only a small number of punctuations. There is a general worry that if the limit size of the sequence is too large (more than two times of the mean or median), the result of the model will be less precise because there are too many harmful useless padding 0 (on the right side). We've thus tried 160, 140 respectively as maximum sequence length and/or use only character and number in the alphabet but the performance didn't show better. The effect is thus not seen in this data.

Beside of learning on two polarity sentiment classes, we evaluated on three classes (with neutral). We consider 800k tweets without emoticons from SESAMm as an approximate for neutral tweets and put it in the

same pool of 1.6M positive and negative Stanford data. Though this is not really a good approximation but it works quite well in the litteratures ([4], [11]). The loss of the precision is hoped to become less important when the data is scaled up massively. The results are shown in figure 6 and 11. In terms of accuracy, we observed only 65% on three classes while it is 55% in precision positive-negative ([9]). This is far from the level that we've seen in two classes and state-of-the-art results (63.3% in precision positive-negative for three classes on Semeval data). This result demonstrate how the task of recognizing just one more class (neutral phrases) is difficult (opposing to vision when more classes could help guide the gradient better). Also, this poor result suggest that the choice of neutral tweets from SESAMm may be not a good choice as subjective tweets tends to have more sentiment than a general tweets. This evoke the same difficulty of constructing the neutral sentences *automatically* in general (prolific sentences without any sentiment).

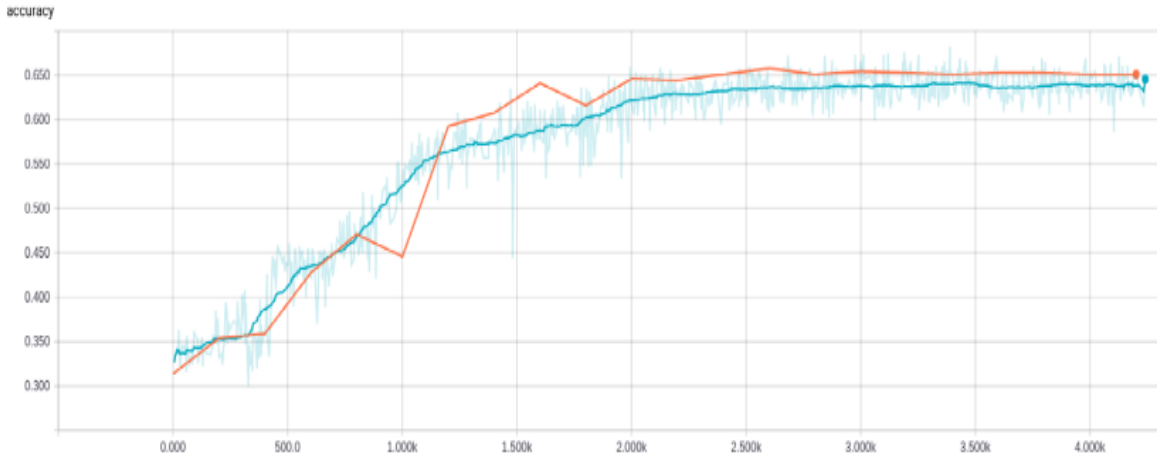


Figure 6: Accuracy of learning on 3 classes

To alleviate the vague notion of neural class in the source data above, we inpsect a broader learning on 9 classes based on emoticons (happy, laughing, kisswink, playful, sad, horror, shock, annoyed, hesitated) on SESAMm dataset. [2]. We hope that this will help the classifier discriminate better and thus generalize well. The learning on test set is as good as on two classes with 85% accuracy performance (figure 7), however result of transfering to 3 classes didn't get any better. It's hard to know if it does not really help or just because we don't have the real hand-labeled samples from each classes (because the emoticons data are somewhat noisy).

4 Conclusion

Transfer learning emerge recently as an important theme of research after the success of deep learning. How to profit a lot of data while still require only a little effort of human-labeling is an interesting subject of study. In this work, we've conducted some preliminary experiments of a deep network on char-level specifically for twitter, where unlabled data is abundant. The distant supervision works pretty well for two classes but it still has a lot of limitations when expanding to three or higher number of classes. Because the constraint of the resource in this study, further works could examine more dataset of emoticons to see if it can have better effect. Also, there's still not an effective way to collect *automatically* neutral sentences in the litterature. Thus, thinking of new method to solve this problem is an important question for the future research.

[2]The classes are chosen from Wikipedia's list: $https://en.wikipedia.org/wiki/List\_of\_emoticons$

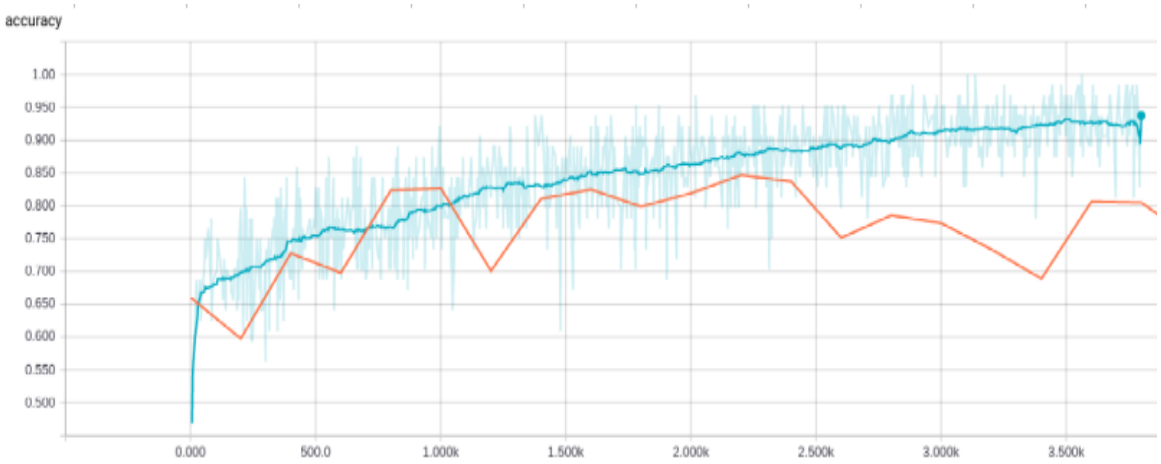


Figure 7: Accuracy of learning on 9 classes

5 Acknowledgement

Part of the experiments realized in this work have been realized on two GPU clusters: Grid5000 (Inria/Loria Nancy) and Romeo Reims (France). The subjective dataset is supported generously by SESAMm. We are very grateful to their owners for giving us access to these resources.

References

[1] A. Conneau, H. Schwenk, L. Barrault, and Y. LeCun. Very deep convolutional networks for natural language processing. *CoRR*, abs/1606.01781, 2016.

[2] J. Deriu, M. Gonzenbach, F. Uzdilli, A. Lucchi, V. D. Luca, and M. Jaggi. Swisscheese at semeval-2016 task 4: Sentiment classification using an ensemble of convolutional neural networks with distant supervision. In *SemEval@NAACL-HLT*, 2016.

[3] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *In Proceedings of the Twenty-eight International Conference on Machine Learning, ICML*, 2011.

[4] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *Processing*, pages 1–6, 2009.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778, 2016.

[6] G. Hinton, L. Deng, D. Yu, G. Dahl, A. rahman Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition. *Signal Processing Magazine*, 2012.

[7] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751, 2014.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[9] P. Nakov, A. Ritter, S. Rosenthal, F. Sebastiani, and V. Stoyanov. Semeval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 1–18, 2016.

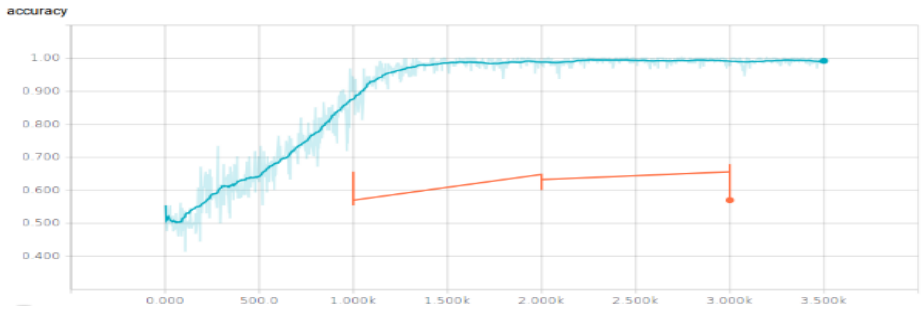
[10] A. Severyn and A. Moschitti. Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 959–962, New York, NY, USA, 2015. ACM.

[11] D. Tang, F. Wei, B. Qin, T. Liu, and M. Zhou. Coooolll: A deep learning system for twitter sentiment classification. In *SemEval@COLING*, 2014.

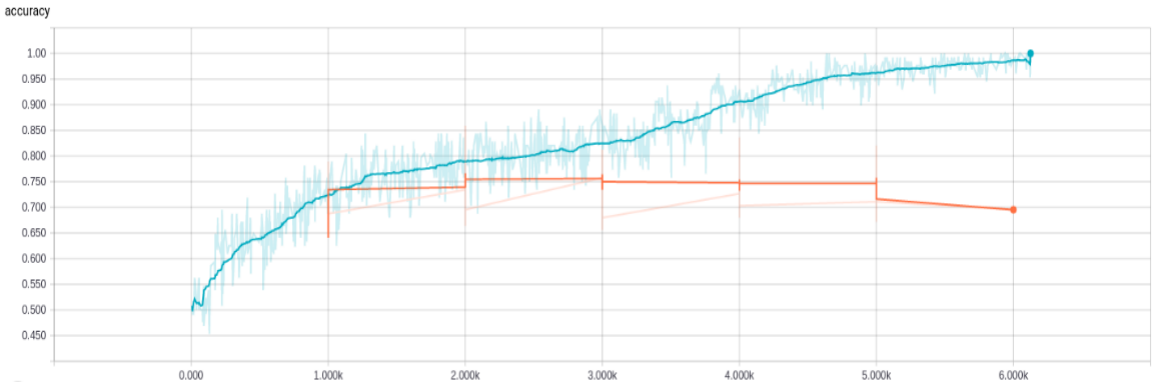
[12] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *CoRR*, abs/1411.1792, 2014.

[13] X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, pages 649–657, Cambridge, MA, USA, 2015. MIT Press.

6 Appendix

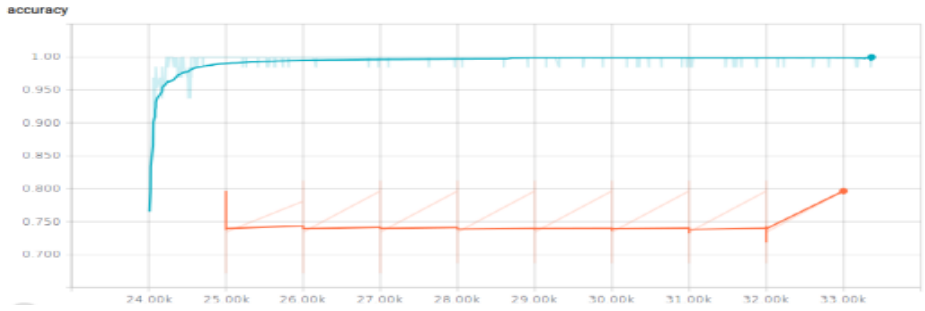


(a) Results on 10k training and test set

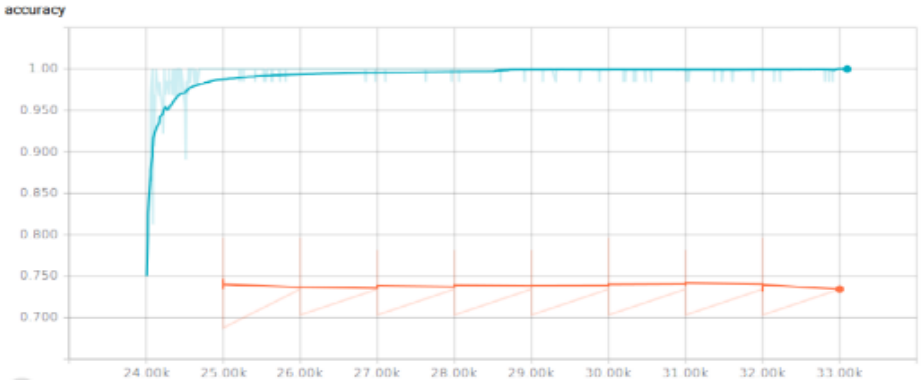


(b) Results on 60k training and test set

Figure 8: Accuracy result on Stanford Emoticon Twitter data



(a) Fine-tune 2 last fully connected layer



(b) Fine-tune 3 last fully connected layer

Figure 9: Accuracy of transfer learning from augmenting Stanford data to SemEval 2013 training and test set, without dropout

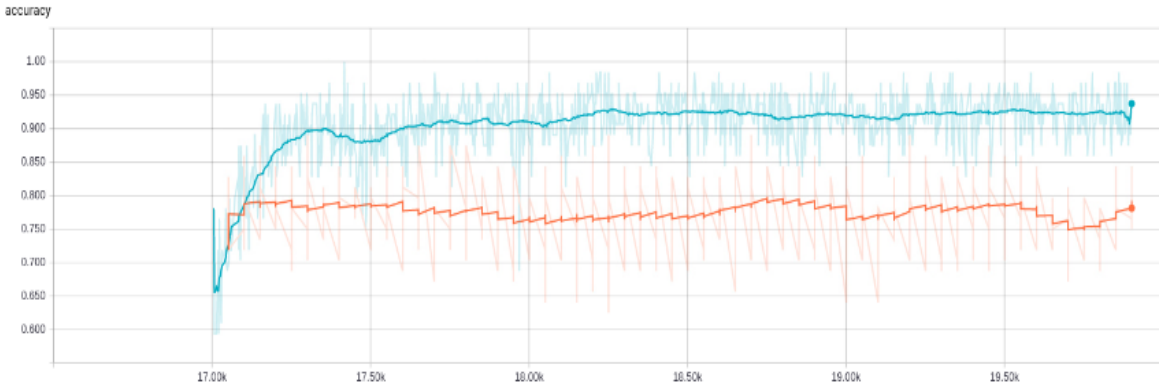
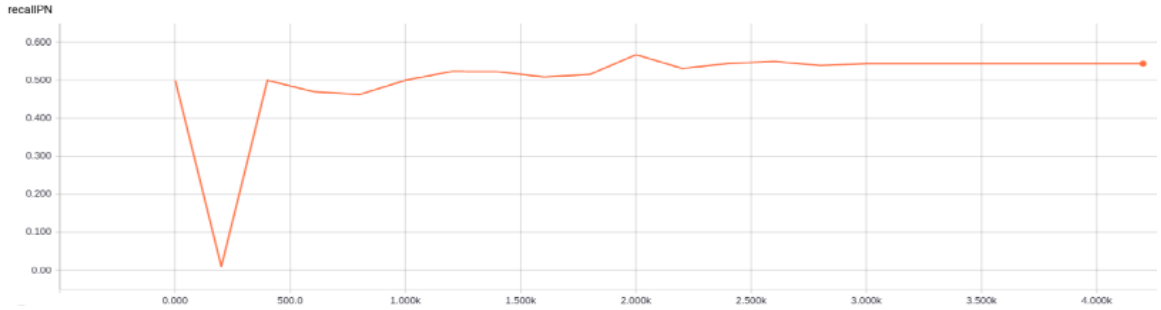


Figure 10: Accuracy of transfer learning from 16M tweets SESAMm data to SemEval 2013 training and test set, with dropout. Fine-tune 2 last fully connected layer



(a) Recall Positive-Negative of learning on 3 classes

(b) F1 Positive-Negative of learning on 3 classes

Figure 11: Results of learning on 3 classes